

# Universidad de Alicante

INGENIERÍA ROBÓTICA

## ROBOTS MÓVILES

*Trabajo final*

*Teleoperación de un dron simulado en Gazebo con Arduino*



Autores:

Martínez Juan, Pablo

Pastor Bernal, Daniel

Enero 2023

# Índice

<b>I   Estado del Arte</b>	<b>2</b>
<b>1. Introducción</b>	<b>3</b>
1.1. Tipología . . . . .	3
<b>2. Teleoperación</b>	<b>4</b>
<b>3. Drones en ROS</b>	<b>5</b>
<b>4. Drones en Gazebo</b>	<b>5</b>
<b>II   Desarrollo</b>	<b>7</b>
<b>5. Desarrollo práctico</b>	<b>8</b>
5.1. Lectura y procesamiento de datos - Arduino . . . . .	8
5.2. Comunicación: Arduino —> ROS . . . . .	11
5.3. Nodo teleoperación - ROS . . . . .	11
<b>6. Simulación</b>	<b>12</b>
<b>7. Repositorio y Guía de uso</b>	<b>14</b>
<b>8. Vídeo demostrativo</b>	<b>14</b>
<b>III   Referencias</b>	<b>15</b>
<b>9. Bibliografía</b>	<b>16</b>

## Parte I

# Estado del Arte

## 1. Introducción

Los drones son una tecnología relativamente nueva que ha tenido un impacto significativo en diversos campos. Estos dispositivos, también conocidos como vehículos aéreos no tripulados (UAV, por sus siglas en inglés), pueden ser controlados de manera remota o programados para volar de forma autónoma.

Los drones se han utilizado en una amplia variedad de aplicaciones, incluyendo la vigilancia, la entrega de paquetes, la agricultura, la construcción y la exploración de áreas de difícil acceso. También se han utilizado en operaciones militares y en la realización de películas y fotografías aéreas.

La tecnología de los drones está en constante evolución, y se espera que su uso siga creciendo en el futuro. Sin embargo, su uso también ha generado controversias y preocupaciones éticas y legales, como la privacidad y la seguridad. Es importante considerar estos problemas a medida que se desarrolla y utiliza esta tecnología.



Figura 1: Dron

### 1.1. Tipología

Existen varios tipos de drones, que se diferencian en función de su tamaño, capacidad y propósito. Algunos ejemplos de tipos comunes de estos incluyen:

- Drones de consola: son pequeños drones diseñados para volar alrededor del interior de un edificio. Suelen ser muy maniobrables y pueden ser controlados con un control remoto o una aplicación móvil.
- Drones de fotografía aérea: son drones diseñados especialmente para tomar fotografías y videos aéreos. Algunos modelos cuentan con cámaras de alta calidad y estabilizadores de imagen para obtener nítidas.

- Drones de entrega: son drones que se utilizan para entregar paquetes y otros objetos de un lugar a otro. Algunos modelos tienen una capacidad de carga limitada, mientras que otros pueden transportar objetos más pesados.
- Drones industriales: son drones utilizados en aplicaciones industriales, como la inspección de líneas de transmisión de energía o la vigilancia de instalaciones industriales.
- Drones militares: son drones diseñados para uso militar, y pueden ser utilizados para actividades como la vigilancia, el reconocimiento y el ataque.
- Drones de exploración: son drones diseñados para explorar áreas de difícil acceso, como cuevas o edificios en ruinas. Pueden ser equipados con cámaras y otros equipos de exploración para enviar imágenes y datos de vuelta a la base.

## 2. Teleoperación

La teleoperación de drones es el proceso de controlar y operar un dron de forma remota, utilizando una consola de control y una conexión de transmisión de datos. Esto permite que el dron sea manejado desde una ubicación diferente a la del dispositivo, lo que amplía significativamente su alcance y capacidad de operación.

Los sistemas de teleoperación de drones suelen incluir una consola de control y una unidad de transmisión de datos, que pueden ser llevados por el operador o instalados en un vehículo o estación fija. La consola de control permite que el operador envíe comandos y reciba datos del dron, mientras que la unidad de transmisión se encarga de enviar y recibir señales de radiofrecuencia entre la consola y el dron.



Figura 2: Consola teleoperación

### 3. Drones en ROS

ROS (Robot Operating System) es un sistema operativo para robots que permite a los usuarios desarrollar, integrar y utilizar una amplia variedad de componentes de software y hardware para robots móviles, manipuladores, sensores y otros dispositivos.



Figura 3: ROS

Los drones también pueden utilizar ROS para facilitar la integración de diferentes componentes de hardware y software y para simplificar el desarrollo de aplicaciones. Algunas de las tareas que pueden realizar los drones con ROS incluyen:

- Navegación autónoma: utilizando datos de sensores y algoritmos de navegación, los drones pueden moverse de forma autónoma a través de un entorno conocido o desconocido.
- Control de vuelo: ROS puede utilizarse para controlar el vuelo de un dron de forma autónoma o semiautónoma, utilizando comandos de velocidad y orientación.
- Procesamiento de datos de sensores: los drones a menudo están equipados con sensores que recogen datos del entorno, como imágenes de cámaras o lecturas de GPS. ROS puede utilizarse para procesar y analizar estos datos para facilitar la toma de decisiones.
- Integración con otros dispositivos: los drones pueden utilizar ROS para integrarse con otros dispositivos, como robots móviles o sistemas de información, para realizar tareas de forma conjunta.

### 4. Drones en Gazebo

Gazebo es un simulador de robots 3D que permite a los usuarios desarrollar, probar y validar sistemas de control y aplicaciones para robots móviles y manipuladores en un entorno virtual. El simulador incluye un motor de física realista que modela la dinámica de movimiento y colisión de los robots y otros elementos del entorno.



Figura 4: Gazebo

Los drones también pueden utilizar Gazebo para probar y validar el control y la navegación autónoma. Algunas de las tareas que pueden realizar los drones en Gazebo incluyen:

- Simulación de vuelo: Gazebo puede utilizarse para simular el vuelo de un dron en un entorno virtual, permitiendo a los usuarios probar diferentes escenarios y sistemas de control sin tener que utilizar un dron real.
- Pruebas de navegación autónoma: los usuarios pueden utilizar Gazebo para probar algoritmos de navegación autónoma en un entorno virtual y evaluar su rendimiento.
- Validación de sistemas de control: Gazebo permite a los usuarios validar y depurar sistemas de control de drones antes de su implementación en el mundo real.
- Integración con otros simuladores: Gazebo puede utilizarse para integrar drones en simulaciones más amplias que involucren otros robots o sistemas, permitiendo evaluar la interacción entre ellos.

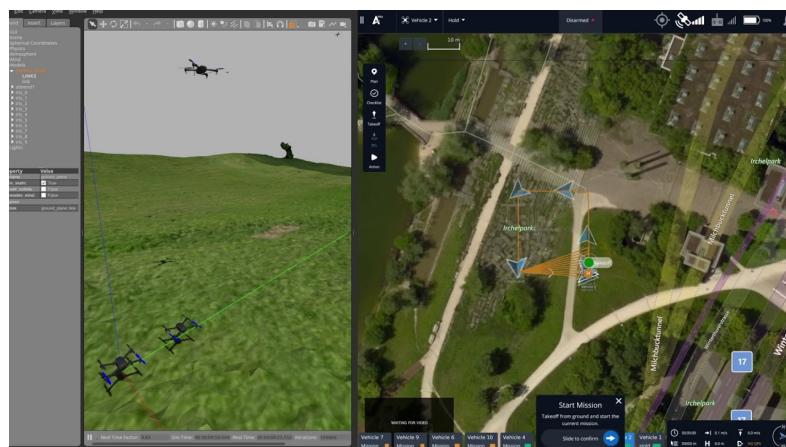


Figura 5: Dron simulado en Gazebo

## Parte II

# Desarrollo

## 5. Desarrollo práctico

Para este trabajo se ha decidido teleoperar un dron mediante dispositivos de bajo coste como el MPU-6050 GY-521, una unidad de medición inercial o IMU (Inertial Measurement Units) de 6 grados de libertad (DoF) que combina un acelerómetro y un giroscopio.

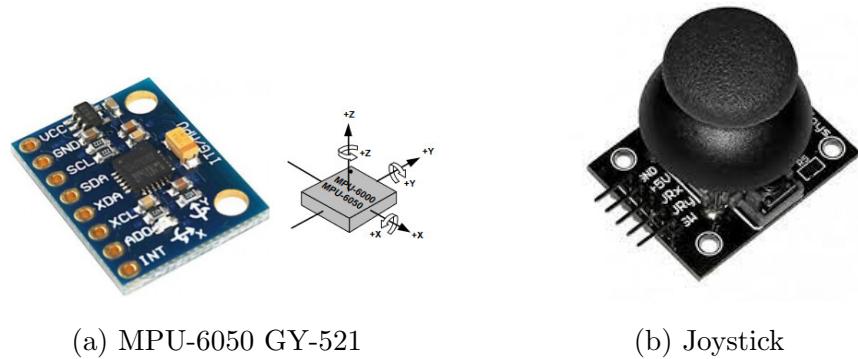


Figura 6: Dispositivos utilizados para la teleoperación

Para ello se conectará la IMU a un Arduino que será el encargado de comunicarse con ROS mediante el uso de nodos, *topics* y mensajes, la forma estándar de comunicación en ROS.

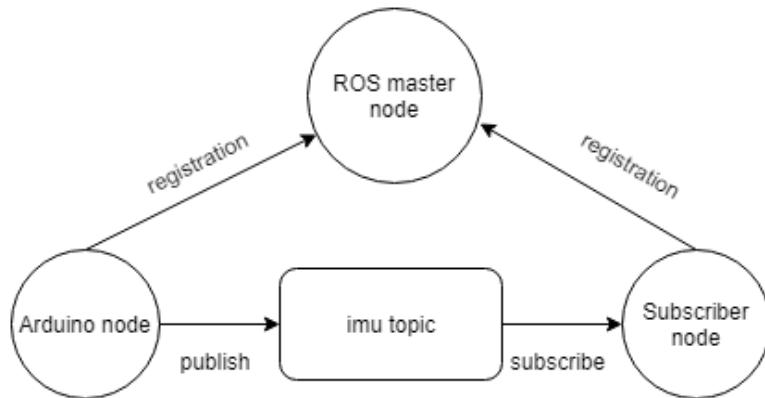


Figura 7: Comunicación

En cuanto a la visualización del dron, se empleará Gazebo, ya que no se dispone de un dron real y este simulador permite generar un entorno realista en el que teleoperar el dispositivo.

### 5.1. Lectura y procesamiento de datos - Arduino

En este punto se va a tratar la medición del ángulo de la IMU conectada a arduino que se usará como ‘volante’. Este dispositivo devuelve valores de aceleración y velocidad angular que

se pueden combinar para obtener, de forma bastante precisa, los ángulos de Euler (roll, pitch, yaw). Remarcar que este dispositivo tiene un coste de 1.16€ en [Aliexpress](#) y a pesar de su bajo coste es posible obtener valores útiles para realizar la teleoperación.

Los ángulos de Euler nos permiten controlar 3 de los 4 movimientos básicos de un dron, guiñada (hacia la derecha o izquierda del eje vertical), inclinación (hacia la derecha o izquierda del eje longitudinal) y cabeceo (rotación hacia delante o hacia atrás con respecto al eje transversal), siendo necesario añadir otro dispositivo para controlar la altitud (elevación en vertical). Para ello se ha decidido utilizar un joystick.

Teniendo claros los elementos que se usarán de la IMU, el siguiente paso será realizar la conexión al un microcontrolador Arduino para proceder a la lectura de datos.

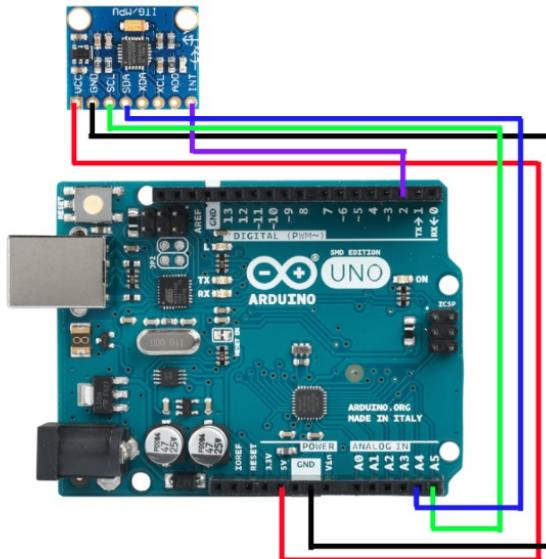


Figura 8: Esquema de conexión IMU

A la hora de la lectura de datos se han empleado diferentes librerías que facilitan este proceso, devolviendo como variable flotante los valores de aceleración y velocidad angular en cada uno de los 3 ejes (X, Y, Z). A estos valores se les aplica un filtro complementario para obtener las rotaciones y reducir el error que se pueda generar en la lectura, ya que el acelerómetro tiene mucho ruido y el giroscopio tiene una deriva en la medida.

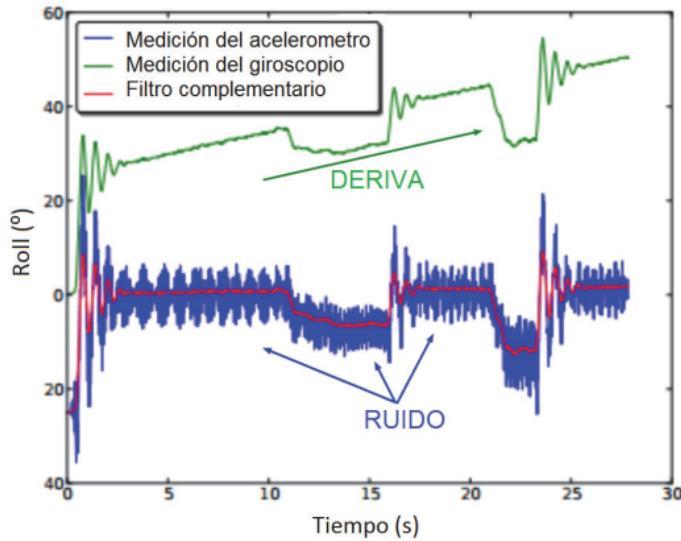


Figura 9: Corrección de errores empleando un filtro complementario

El filtro complementario incluye los filtros digitales paso-bajo para el acelerómetro y paso-alto para el giroscopio, y además lleva a cabo la combinación ponderada de las medidas de los dos dispositivos. Todo ello con operaciones simples, que consumen pocos recursos, necesario a la hora de utilizar microcontroladores de bajo coste, como en este caso.

Para la conexión del Joystick a la placa se pueden seguir las siguientes indicaciones. De tal manera que se conecte a 2 entradas analógicas, correspondientes a los ejes, y a una digital que represente el botón que incorpora.

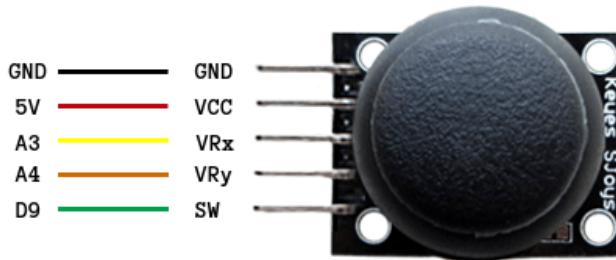


Figura 10: Conexionado Joystick

En cuanto a la lectura de datos del joystick, se realiza de una sencilla manera puesto que proporciona 2 valores analógicos representando la posición en cada uno de sus ejes (X e Y). No requieren ningún tipo de procesamiento ni filtrado inicial puesto que son bastante precisos. La recepción de la pulsación tampoco se ha de procesar.

## 5.2. Comunicación: Arduino —> ROS

Una vez obtenidos los valores necesarios para la teleoperación del dron, tanto de la IMU como del joystick, se deben comunicar estos datos a ROS. Para ello se ha empleado la librería *ros.h* que permite crear *nodos* y acceder a *topics* desde Arduino. Se ha creado un nodo *publisher* y el *topic* ‘imu’ donde se publica el mensaje ‘imu.msg’ de tipo *String*.

Para el envío del mensaje se ha empleado una cadena de caracteres, concatenando las diferentes variables separadas por letras como se muestra a continuación.

```
String data = "A" + AX + "B" + AY + "C" + Joy + "D" + But + "E";
```

Listing 1: Codificación mensaje en C++

Se ha empleado esta forma ya que consume pocos recursos del microcontrolador y es muy sencilla de decodificar mediante código en *python*, solo es necesario seleccionar el fragmento de la cadena que se desea guardar en cada variable. Para ello se indica el índice de las letras empleadas en la codificación como se observa a continuación.

```
Ax = float(imu.data[imu.data.index('A')+1:imu.data.index('B')])
```

Listing 2: Decodificación mensaje en *python*

De esta sencilla forma es posible almacenar cada valor en una variable para poder continuar con la gestión de los datos desde el nodo de ROS encargado de la teleoperación.

## 5.3. Nodo teleoperación - ROS

Una vez la información ha sido recibida a través del microcontrolador, procesada para enviarse y publicada en su *topic* correspondiente, se accede a ella desde otro nodo ROS. La principal función de esta parte es el procesado final de la información física con el fin de obtener sus correspondencias en velocidades para aplicar al dron.

En este nodo se definen dos partes diferenciadas. Una encarga de la decodificación de los datos, de la manera comentada con anterioridad, y otra del cálculo y publicación de velocidades en el

*topic* correspondiente del dron.

Es destacable que las velocidades aplicadas se calculan en función de unas ganancias proporcionales predefinidas, con el fin de que a mayores inclinaciones de la IMU, o movimientos del Joystick, el dron se mueva más rápido.

También cabe mencionar los 2 modos de funcionamiento que se han implementado. Pudiendo cambiar en tiempo real entre ellos pulsando el botón incorporado en el Joystick.

- **No holonómico** -> En este modo, el dron solo avanza (o retrocede) en la dirección del eje X local definido en su base, aplicando en ella velocidad lineal. Es decir, para realizar giros se ha de rotar el dron sobre su eje Z, esto quiere decir que las velocidades de giro asignadas solo van a ser angulares sobre dicho eje.
- **Holonómico** -> En este otro, el dron tiene la capacidad de avanzar en cualquier dirección dentro de sus grados de libertad, velocidades lineales en todos los ejes. Es decir, sin necesidad de rotar y direccionar la base.

En ambos modos se gestiona de la misma manera la elevación del dron, aplicando velocidades lineales al eje Z de la base. Si se desea, se puede acceder a una visualización de los modos de vuelo más detallada. En ella se pueden observar las diferencias, como que en el no holonómico gira la cámara montada en el dron, mientras que en el otro no lo hace.

## 6. Simulación

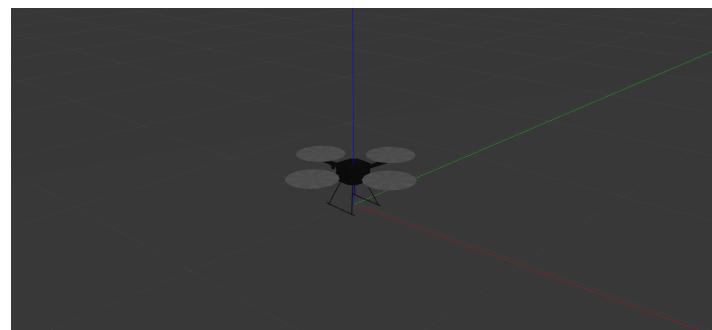
Tras diversas pruebas, para la simulación del dron se ha escogido el paquete de ROS hector-quadrotor-noetic. Este permite simularlo en diferentes mundos, acceder a sus topics de velocidad, cámara, etc.



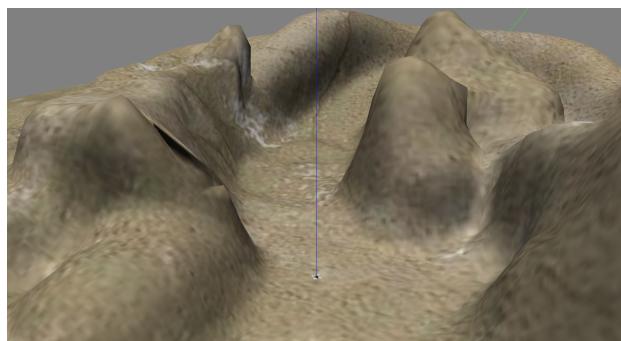
Figura 11: Dron *Hector Quadrotor*

Cabe destacar que en el apartado 7 se encuentra el enlace al repositorio del proyecto, por lo que no se va a entrar en detalle acerca de cómo simular. En dicho repositorio se puede acceder a una guía de uso detallada. Es más, en el apartado 8 se puede acceder a un tutorial en el que se siguen las indicaciones del repositorio en tiempo real y se ejecutan diversas simulaciones.

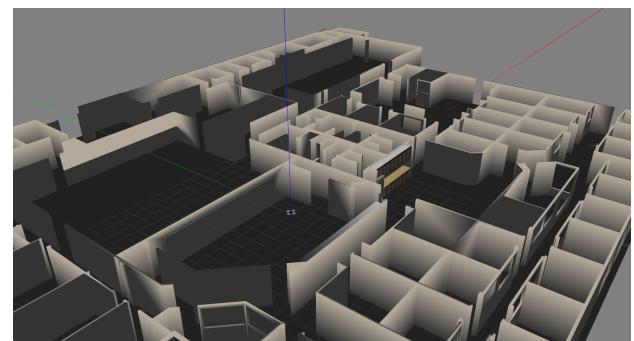
En las simulaciones se ha lanzado el dron en diferentes mundos ya predefinidos, contenidos en el paquete. Inicialmente se ha probado que la instalación haya sido correcta, simulando tan solo el UAV en un mundo vacío, posteriormente ya se han cargado otros mundos.



(a) Mundo vacío



(b) Mundo exterior



(c) Mundo interior

Figura 12: Dron simulado en diferentes mundos en Gazebo

Incluso se ha personalizado un mundo en el que se ha cargado el modelo del dron, con el fin de observar su comportamiento y teleoperarlo en un espacio fuera de los que incorpora por defecto el paquete.



Figura 13: Dron simulado en mundo personalizado

## 7. Repositorio y Guía de uso

Con el fin de mantener un control de versiones y poder tener acceso a todo el código actualizado por parte de los integrantes del grupo, se creó un [Repositorio GitHub](#). En él, se pueden encontrar las diferentes partes del proyecto, el trabajo en Arduino y el paquete de teleoperación, así como una guía de usuario para poder ejecutar los procesos.

## 8. Vídeo demostrativo

También se ha grabado un tutorial de uso, siguiendo las indicaciones del repositorio comentado anteriormente. De tal manera que se pueda llegar a simular y teleoperar el dron si se desea.



Figura 14: Pulse la imagen para acceder al vídeo

## Parte III

# Referencias

## 9. Bibliografía

### Referencias

- [1] Tipología Drone
- [2] Drones: qué son, cómo funcionan, normativa Española
- [3] ¿Qué es un DRONE? Tipos, nombres y componentes
- [4] TUTORIAL MPU6050, ACELERÓMETRO Y GIROSCOPIO
- [5] Ángulos De Rotación Con El IMU De Arduino Nano 33 IoT
- [6] DETERMINAR LA ORIENTACIÓN CON ARDUINO Y EL IMU MPU-6050
- [7] Measure Pitch Roll and Yaw Angles Using MPU6050 and Arduino
- [8] ROS, IMU and an Arduino: How to read IMU sensor output and send it to ROS